

Міністерство освіти і науки України
Національний технічний університет України
«Київський політехнічний інститут імені Ігоря Сікорського»

ТЕОРІЯ АЛГОРИТМІВ

**МЕТОДИЧНІ ВКАЗІВКИ
ДО ПРАКТИЧНИХ РОБІТ**

КИЇВ 2018

Теорія алгоритмів. Методичні вказівки до практичних робіт. / Уклад.:
Т.А.Ліхоузова – К.: КПІ ім. Ігоря Сікорського, 2018. – 27 с.

Посібник призначений для студентів спеціальностей 121 «Інженерія програмного забезпечення» та 126 «Інформаційні системи та технології» всіх форм навчання.

Укладач

Ліхоузова Т.А., к.т.н., доцент

Відповідальний редактор

Ткач М.М., к.т.н., доцент

Рецензент

Букасов М.М., к.т.н., доц.,
доцент кафедри АУТС ФІОТ

За редакцією укладачів

ПРАКТИЧНА РОБОТА №1

Тема: Побудова і аналіз алгоритмів

Мета роботи: ознайомитись з роботами, що виконує програміст на кожному з етапів розв'язку задачі.

Запитання на допуск до роботи

1. З яких етапів складається процес створення комп'ютерної програми для вирішення довільної практичної задачі?
2. Що саме має з'ясувати розробник програми на етапі постановки задачі?
3. Що робить розробник програми на етапі побудови моделі? Які фактори впливають на вибір структури моделі?
4. Якими міркуваннями має керуватися розробник програми на етапі розробки алгоритму? Чи потрібно перевіряти або доводити правильність алгоритму, якщо так, то з якою метою?
5. Перерахуйте основні складності, з якими стикається розробник на етапі реалізації алгоритму у вигляді програми.
6. Навіщо виконується аналіз алгоритму та його складності? Перерахуйте основні прийоми аналізу.
7. Існує три аспекти перевірки програми: на правильність, на ефективність реалізації, на обчислювальну складність. Розкрийте суть кожної з перевірок.
8. Для чого служить документація програми? Яких різновидів вона буває? На якому етапі розробки програми виконується?
9. Навіщо виконується вимірювання часу виконання програми? Які чинники на нього впливають?

Завдання

Виконати всі етапи розробки програми.

Постановка задачі – чітко вказати, що дано і що треба знайти.

Побудова моделі – які структури даних та які математичні залежності використані.

Розробка алгоритму – опис алгоритму у вигляді блок-схеми.

Правильність алгоритму – довести покроково правильність розробленого алгоритму.

Аналіз алгоритму та його складності – оцінити використовуючи O -символіку час виконання алгоритму в найгіршому або/і в середньому.

Реалізація алгоритму – навести текст програми.

Перевірка програми – описати тестові дані для перевірки програми на правильність, ефективність реалізації та обчислювальну складність (для цього використайте блок-схему алгоритму та профілі виконання програми).

Результатом практичної роботи є повна документація по програмі.

Варіанти завдань

- 1) Пошук мінімального елемента (індекс і значення) в одномірному масиві з n елементів.
- 2) Пошук мінімального елемента (індекс і значення) в двомірному масиві з $n \times m$ елементів.
- 3) В одномірному масиві з n елементів поміняти місцями максимальний та мінімальний елементи.
- 4) В одномірному масиві з n елементів обчислити кількість елементів більше заданого числа.
- 5) В двомірному масиві з $n \times m$ елементів знайти кількість нулів.
- 6) В двомірному масиві з $n \times n$ елементів знайти суму діагональних елементів.

Масиви заповнити випадковими числами автоматично. В програмі передбачити наступні варіанти отримати вхідні дані для обчислень: ввести вручну, згенерувати автоматично (і записати ці дані у файл); прочитати з файлу.

Приклад звіту з практичної роботи №1.

Тема: Побудова і аналіз алгоритмів.

Постановка задачі

Спроектувати і реалізувати програму розрахунку таблиці відповідностей між мірами. Початкове значення міри, крок зміни цього значення і кількість рядків у таблиці задати в режимі діалогу.

1 доба = 24 години = 1440 хвилин

Побудова моделі

Основні величини:

d	цілий або дійсний тип	кількість діб
g	цілий або дійсний тип	кількість годин
h	дійсний тип	кількість хвилин
k	цілий або дійсний тип	крок зміни діб
n	цілий додатній тип	кількість рядків в таблиці

Допоміжні величини:

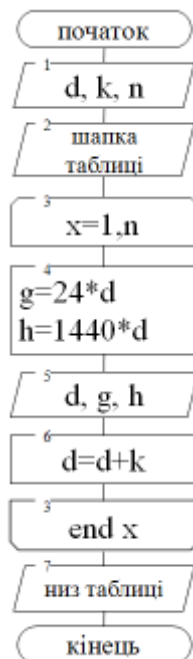
x	цілий тип	лічильник рядків в таблиці
---	-----------	----------------------------

Використані математичні залежності:

$$g = 24 \cdot d$$

$$h = 1440 \cdot d = 60 \cdot g$$

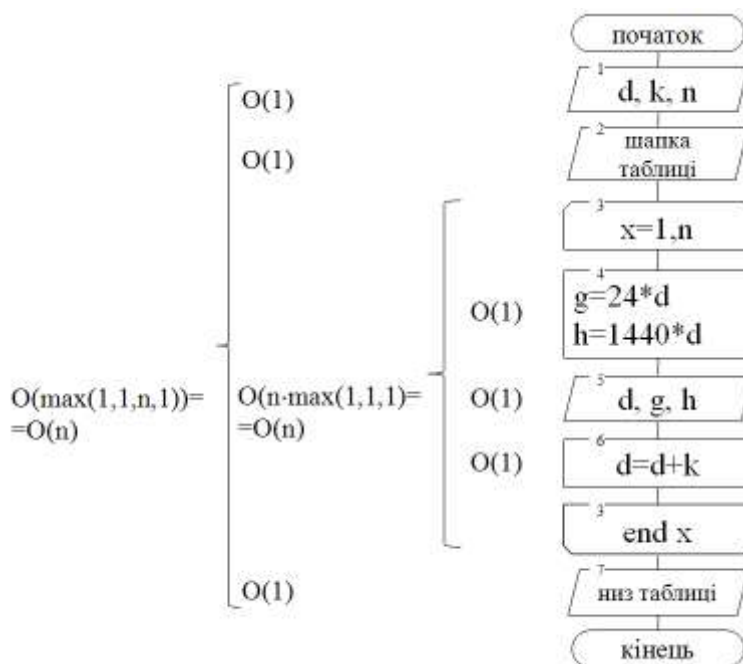
Розробка алгоритму



Правильність алгоритму

1. Обґрунтування правомірності кожного кроку: блок 1 відповідає за введення початкових даних – без них алгоритм не працюватиме; блоки 2 і 7 призначені для зручного перегляду результатів; блок 3 забезпечує виведення потрібної кількості результатів; блоки 4 і 6 забезпечують обчислення результатів, а блок 5 – їх виведення на екран.
2. Доведення кінцевості алгоритму: кінцевість алгоритму залежить тільки від блоків 3, 4 і 6; початкове значення змінної x завжди менше або рівне кінцевому значенню n лічильника циклу (обумовлено типом n), в тілі циклу ніякого впливу на змінні x та n немає, отже кількість ітерацій циклу кінцева і дорівнює n .

Аналіз складності алгоритму



Загальна складність алгоритму $O(n)$.

Реалізація алгоритму

// розрахунок таблиці відповідностей між мірами

// розробник Іванов О.В.

```
#include "stdafx.h"
```

```
#include "stdio.h"
```

```
#include <conio.h>
```

```

void main()
{
    // ініціалізація початкових значень
    float d = 0, k = 0, g = 1, h = 1;
    int n = 1, x = 1;

    // введення вхідних даних
    printf("input d\n");
    scanf_s("%f", &d);
    printf("input k\n");
    scanf_s("%f", &k);
    printf("input n\n");
    scanf_s("%d", &n);

    // формування та виведення таблиці
    printf(" days \t \t hours \t \t seconds \n-----\n");
    for(x = 0; x != n; x++)
    {
        g = 24 * d;
        h = 1440 * d;
        printf("%f\t%f\t%f\n", d, g, h);
        d += k;
    }
    printf("-----\n");
    getch();
}

```

Перевірка правильності програми

Для перевірки використаємо три типи вхідних даних: детерміновані, випадкові та некоректні.

Вхідні дані			Результат	Призначення тесту
d	k	n		
1	1	1	відомий	перевірка використаних математичних залежностей
3	2 -2 0,5	10	відомий	перевірка кінцевості розрахунків для різних кроків зміни діб
випад-кове	випад-кове	випад-кове	невідомий, але легко обчислити	перевірка ефективності реалізації та обчислювальної складності алгоритму із замірюванням часу виконання програми та побудовою профілю програми.
*	*	*	відомий	перевірка реакції програми на некоректні вхідні дані (додавання повідомлень про помилки)

Перевірка обчислювальної складності програми



Перевірка обчислювальної складності програми підтвердила аналітичну оцінку $O(n)$.

ПРАКТИЧНА РОБОТА №2

Тема: Методи розробки алгоритмів

Мета роботи: порівняння алгоритмів розв'язку задачі, побудованих різними методами.

Запитання на допуск до роботи

1. Перерахуйте відомі вам методи розробки алгоритмів. Докладніше розкажіть про один з них.
2. Перерахуйте переваги та недоліки наступних методів розробки алгоритмів: методу часткових (проміжних) цілей, методу підйому (локального пошуку), методу відпрацювання назад.

3. Який тип алгоритмів називають «жадібними» і чому?
4. Дайте характеристику евристичним алгоритмам. В яких випадках доцільно використовувати цей тип алгоритмів? Опишіть загальний підхід до побудови евристичних алгоритмів.
5. Проаналізуйте, що спільного мають та чим відрізняються алгоритми, що використовують пошук з поверненням, та алгоритми, що використовують метод гілок та границь.
6. Поясніть, для чого можна використовувати метод альфа-бета відсікання.
7. Поясніть термін «структурне програмування». Для чого воно застосовується?
8. Методологія структурного програмування ґрунтується на трьох методах: низхідного проектування, модульного програмування і структурування програм. Розкрийте суть методу низхідного проектування.
9. Методологія структурного програмування ґрунтується на трьох методах: низхідного проектування, модульного програмування і структурування програм. Розкрийте суть методу модульного програмування.
10. Методологія структурного програмування ґрунтується на трьох методах: низхідного проектування, модульного програмування і структурування програм. Розкрийте суть методу структурування програм.

Завдання

Ознайомтесь з алгоритмами розв'язку задачі комівояжера, побудованих різними методами (метод проміжних цілей 298 [1], 114 [2], метод повного перебору 296 [1], 20, 132 [2], метод локального пошуку 303 [1]).

Розробіть або знайдіть в літературі алгоритми розв'язку задачі свого варіанту (мінімум двома методами). Побудуйте блок-схеми цих алгоритмів (деталізація схем повинна бути достатньою для можливості виконання оцінки складності алгоритму). Проаналізуйте ефективність та аналітичну складність алгоритмів і вкажіть переваги та недоліки кожного з них.

Варіанти завдань

- 1) Фарбування вершин графу.

- 2) Каркас мінімальної ваги.
- 3) Найкоротший шлях між заданими вершинами.
- 4) Найбільша течія в мережі між заданими вершинами.

Додаткове завдання

Напишіть програмні реалізації алгоритмів. Виміряйте час виконання програм для різних реалізацій на одному наборі тестових прикладів. Порівняйте результати з аналітично отриманою обчислювальною складністю відповідних алгоритмів.

Склад звіту практичної роботи

- постановка задачі для кожного з методів;
 - блок-схеми, на яких виконано аналіз складності алгоритмів;
 - висновки про доцільність використання кожного з алгоритмів;
- якщо виконано додаткове завдання:
- опис тестових даних (якого характеру дані і для якої перевірки використані);
 - результати дослідження у вигляді графіків або діаграм;
 - висновки про доцільність використання кожного з алгоритмів для типових вхідних даних та про відповідність результатів експериментального дослідження аналітичним оцінкам складності.

ПРАКТИЧНА РОБОТА №3

Тема: Абстрактні типи даних

Мета роботи: порівняння реалізацій абстрактних типів даних, побудованих на базі різних структур даних.

Запитання на допуск до роботи

1. Поясніть різницю між термінами «тип даних», «абстрактний тип даних» «структура даних». Для чого вони застосовується?
2. Проаналізуйте АТД «зв'язний лінійний список»; перерахуйте, які різновиди списків бувають та прокоментуйте та порівняйте їх способи реалізації.

3. Проаналізуйте АТД «стек»; перерахуйте, які операції можна виконувати з цим АТД, прокоментуйте та порівняйте способи реалізації АТД.
4. Перерахуйте та проілюструйте послідовність дій при вставці елемента до стеку.
5. Перерахуйте та проілюструйте послідовність дій при видаленні елемента зі стеку.
6. Проаналізуйте АТД «черга»; перерахуйте, які операції можна виконувати з цим АТД, прокоментуйте та порівняйте способи реалізації АТД.
7. Перерахуйте та проілюструйте послідовність дій при вставці елемента до черги.
8. Проаналізуйте АТД «однозв'язний лінійний список»; перерахуйте, які операції можна виконувати з цим АТД, прокоментуйте та порівняйте способи реалізації АТД.
9. Перерахуйте та проілюструйте послідовність дій при вставці елемента в середину списку.
10. Перерахуйте та проілюструйте послідовність дій при видаленні елемента з середини списку.
11. Перерахуйте та проілюструйте послідовність дій при видаленні елемента з кінця списку.
12. Проаналізуйте АТД «відображення»; перерахуйте, які операції можна виконувати з цим АТД, прокоментуйте та порівняйте способи реалізації АТД.
13. Наведіть визначення та приклад рекурсії. Як її можна використати в математиці та в програмуванні?
14. Проаналізуйте АТД «дерево»; перерахуйте, які операції можна виконувати з цим АТД, прокоментуйте та порівняйте способи реалізації АТД.
15. Прокоментуйте та порівняйте способи реалізації бінарних дерев.
16. Поясніть термін «обхід дерева». Перерахуйте способи обходу та форми, які будуть отримані в результаті використання кожного із способів.

17. Чим відрізняються дерева бінарного пошуку та для чого вони використовуються?

18. Перерахуйте та проілюструйте послідовність дій при додаванні вузла в бінарне дерево пошуку.

19. Перерахуйте та проілюструйте послідовність дій при видаленні вузла з бінарного дерева пошуку.

Завдання

Ознайомтесь з реалізаціями абстрактних типів даних за допомогою масивів та покажчиків (23-99 [1], 310-328 [4]). Для одного з АТД побудуйте блок-схеми алгоритмів для додавання та видалення елементів, використовуючи дві різні реалізації (за допомогою масивів та покажчиків). Напишіть програмні реалізації алгоритмів. Порівняйте результати експериментів по оцінці ефективності реалізації з аналітичною оцінкою складності алгоритмів. Проаналізуйте ефективність та складність алгоритмів і вкажіть переваги та недоліки кожної з реалізацій.

Варіанти завдань

- 1) АТД «черга»
- 2) АТД «однозв'язний лінійний список»
- 3) АТД «двозв'язний лінійний список»
- 4) АТД «відображення»

Склад звіту практичної роботи

- постановка задачі (вказати, який АТД досліджується, та які реалізації вибрано);
- блок-схеми реалізацій, на яких виконано аналіз складності алгоритмів (розглянути тільки операції додавання та видалення елемента);
- опис тестових даних (якого характеру дані і для якої перевірки використані);
- результати дослідження у вигляді графіків або діаграм;

- висновки про доцільність використання кожної з реалізацій для типових вхідних даних та про відповідність результатів експериментального дослідження аналітичним оцінкам складності.

ПРАКТИЧНА РОБОТА №4

Тема: Пошук у масиві. Відкрите хешування

Мета роботи: дослідження схем відкритого хешування.

Запитання на допуск до роботи

1. Сформулюйте та поясніть термін «задача пошуку».
2. Розкрийте суть абстрактного типу даних «Словник». Які операції включено в цей тип даних?
3. Поясніть, виходячи з чого обирається конкретна реалізація абстрактного типу даних «Словник» для розв'язку задачі та перерахуйте можливі варіанти реалізації.
4. Перерахуйте та прокоментуйте переваги та недоліки реалізації словника за допомогою:
 - сортованих або несортованих зв'язних списків.
 - двійкового вектора (масиву, індексованого по ключам).
 - масиву фіксованої довжини з покажчиком на останній заповнений елемент.
 - масиву з використанням хешування.
 - бінарного пошуку.
5. Поясніть термін «хешування». Перерахуйте та прокоментуйте переваги та недоліки форм хешування.
6. Розкрийте на прикладі основну ідею відкритого хешування.

Завдання

Ознайомтесь з реалізаціями схем відкритого хешування (дві хеш-функції для відкритого хешування 122-124 [1], додаткова література [3, розділ 14]). Напишіть програмні реалізації двох хеш-функцій для відкритого хешування.

Проаналізуйте їх ефективність та складність і вкажіть переваги та недоліки кожної з реалізацій.

Склад звіту практичної роботи

- постановка задачі (вказати, які хеш-функції вибрано);
- розглянути тільки операції вдалого пошуку та невдалого пошуку (додавання) елемента. Результати дослідження у вигляді графіків окремо для вдалого та невдалого пошуку – дві хеш-функції в одній координатній сітці, по осі абсцис – поточна відносна кількість елементів у таблиці (відсоток заповнення таблиці), по осі ординат – середня довжина списків (або середній час на виконання операції. Вхідні дані можна генерувати за допомогою випадкових чисел, але в одному досліді один і той самий набір використати для обох хеш-функцій. Обов'язково провести 3-5 серій експериментів з різними вхідними даними і результати усереднити.
- висновки про те, яка з розглянутих хеш-функцій краща, та при якому відсотку заповнення таблиці потрібно проводити її реструктуризацію.

ПРАКТИЧНА РОБОТА №5

Тема: Пошук у масиві. Закрите хешування

Мета роботи: дослідження схем закритого хешування.

Запитання на допуск до роботи

1. Що таке хеш-функція?
2. Перерахуйте порядок дій при закритому хешуванні.
3. Яку ситуацію називають колізією при використанні закритого хешування? Перерахуйте методики вирішення колізій.

Завдання

Ознайомтесь з реалізаціями схем закритого хешування (дві методики вирішення колізій для закритого хешування 120-122, 124-127 [1]; додаткова література [3, розділ 14]). Напишіть програмні реалізації для двох методик

вирішення колізій для закритого хешування. Проаналізуйте їх ефективність та складність і вкажіть переваги та недоліки кожної з реалізацій.

Склад звіту практичної роботи

- постановка задачі (вказати, яку хеш-функцію та методики вирішення колізій вибрано);
- розглянути тільки операції вдалого пошуку та невдалого пошуку (додавання) елемента. Результати дослідження у вигляді графіків окремо для вдалого та невдалого пошуку – дві хеш-функції в одній координатній сітці; по осі абсцис – поточна відносна кількість елементів у таблиці (відсоток заповнення таблиці), по осі ординат – середня кількість колізій (або середній час на виконання операції). Вхідні дані можна генерувати за допомогою випадкових чисел, але в одному досліді один і той самий набір використати для обох хеш-функцій. Обов'язково провести 3-5 серій експериментів з різними вхідними даними і результати усереднити.
- висновки про те, яка з розглянутих хеш-функцій краща, та при якому відсотку заповнення таблиці потрібно проводити її реструктуризацію.

ПРАКТИЧНА РОБОТА №6

Тема: Дерева бінарного пошуку

Мета роботи: дослідження характеристик продуктивності дерев пошуку.

Запитання на допуск до роботи

1. Чим відрізняються дерева бінарного пошуку та для чого вони використовуються?
2. Перерахуйте та проілюструйте послідовність дій при додаванні вузла в бінарне дерево пошуку.
3. Перерахуйте та проілюструйте послідовність дій при видаленні вузла з бінарного дерева пошуку.
4. Прокоментуйте характеристики продуктивності дерев бінарного пошуку на конкретних прикладах.

Завдання

Ознайомтесь з різновидами дерев пошуку (BST-дерева с.498 [3], 310 [4]; розширені BST-дерева с.533 [3]; рандомізовані дерева с.527 [3]; червоно-чорні дерева с.545 [3]). Напишіть програмну реалізацію BST-дерева (передбачити операції додавання, виведення на екран та пошуку). Проведіть серію експериментів для оцінки продуктивності пошуку. Виявіть фактори, що впливають на продуктивність пошуку найбільше (кількість елементів, висота дерева, збалансованість). Порівняйте отримані результати з таблицями на с.489 [3] та с.562-565 [3].

Склад звіту практичної роботи

- постановка задачі (вказати, які дерева досліджуються);
- блок-схеми реалізацій, на яких виконано аналіз складності алгоритмів (розглянути тільки операції вдалого пошуку та невдалого пошуку (додавання) елемента);
- результати дослідження у вигляді графіків для вдалого та невдалого пошуку – дві криві в одній координатній сітці, по осі абсцис – поточна кількість елементів, по осі ординат – середня довжина списків (або середній час на виконання операції). Обов'язково провести 3-5 серій експериментів з різними вхідними даними і результати усереднити.
- висновки про фактори, що впливають на продуктивність пошуку найбільше (кількість елементів, висота дерева, збалансованість).

ПРАКТИЧНА РОБОТА №7

Тема: Збалансовані дерева бінарного пошуку

Мета роботи: дослідження характеристик продуктивності дерев пошуку.

Запитання на допуск до роботи

1. Який тип дерев називають збалансованими деревами пошуку? Перерахуйте підходи до рішення задачі стабілізації продуктивності для реалізацій, заснованих на використанні BST-дерев.

2. Поясніть термін «ротація» та вкажіть, для чого вона використовується. Перерахуйте різновиди ротацій.
3. Перерахуйте та прокоментуйте переваги та недоліки розширених дерев пошуку. Поясніть основні принципи їх побудови.
4. Перерахуйте та прокоментуйте переваги та недоліки рандомізованих дерев. Поясніть основні принципи їх побудови.

Завдання

Модифікуйте програму з попередньої роботи, додавши можливість примусового балансування дерева. Проведіть серію експериментів для оцінки продуктивності пошуку в збалансованому дереві та порівняйте отримані результати з результатами попередньої роботи. За результатами експериментів зробіть висновки: як часто варто робити балансування дерева; для яких задач та даних досліджувані дерева підходять якнайкраще, а у яких випадках використовувати їх недоцільно.

Склад звіту практичної роботи

- постановка задачі (вказати, які дерева досліджуються);
- результати дослідження у вигляді графіків для вдалого та невдалого пошуку – дві криві в одній координатній сітці, по осі абсцис – поточна кількість елементів, по осі ординат – середня довжина списків (або середній час на виконання операції). Обов'язково провести 3-5 серій експериментів з різними вхідними даними і результати усереднити.
- окремо оцініть час, необхідний для балансування дерев різного розміру.
- висновки про доцільність використання досліджених дерев.

ПРАКТИЧНА РОБОТА №8

Тема: Червоно-чорні дерева пошуку

Мета роботи: дослідження характеристик продуктивності дерев пошуку.

Запитання на допуск до роботи

1. Перерахуйте та прокоментуйте переваги та недоліки 2-3-4-дерев. Поясніть основні принципи їх побудови.
2. Перерахуйте та прокоментуйте переваги та недоліки червоно-чорних дерев. Поясніть основні принципи їх побудови.
3. Перерахуйте та прокоментуйте переваги та недоліки навантажених дерев. В яких випадках їх використання найефективніше?

Завдання

Напишіть програмну реалізацію червоно-чорного дерева (передбачити операції додавання, виведення на екран та пошуку). Проведіть серію експериментів для оцінки продуктивності пошуку та порівняйте отримані результати з результатами попередньої роботи. Виявіть фактори, що впливають на продуктивність пошуку найбільше (кількість елементів, висота дерева, збалансованість). За результатами експериментів зробіть висновки, для яких задач та даних досліджувані дерева підходять якнайкраще, а у яких випадках використовувати їх недоцільно.

Склад звіту практичної роботи

- постановка задачі (вказати, які дерева досліджуються);
- блок-схеми реалізацій, на яких виконано аналіз складності алгоритмів (розглянути тільки операції вдалого пошуку та невдалого пошуку (додавання) елемента);
- результати дослідження у вигляді графіків для вдалого та невдалого пошуку – дві криві в одній координатній сітці, по осі абсцис – поточна кількість елементів, по осі ординат – середня довжина списків (або середній час на виконання операції). Обов'язково провести 3-5 серій експериментів з різними вхідними даними і результати усереднити.
- висновки про доцільність використання кожної з реалізацій для типових вхідних даних та про фактори, що впливають на продуктивність пошуку найбільше (кількість елементів, висота дерева, збалансованість).

ПРАКТИЧНА РОБОТА №9

Тема: Алгоритми сортування. Елементарні методи сортування

Мета роботи: дослідження та порівняння алгоритмів сортування.

Запитання на допуск до роботи

1. Сформулюйте задачу сортування.
2. Поясніть, виходячи з чого обирається алгоритм сортування для розв'язку конкретної задачі.
3. Розкрийте класифікацію методів сортування за принципом роботи (адаптивні та не адаптивні методи).
4. Розкрийте класифікацію методів сортування за потребою додаткового обсягу оперативної пам'яті, що використовується алгоритмом сортування.
5. Розкрийте класифікацію методів сортування за стійкістю.
6. Розкрийте класифікацію методів сортування за способом доступу до елементів (прямі та непрямі методи).

Завдання

Ознайомтесь з різновидами елементарних методів сортування (с.249-298 [3]).

Напишіть програмні реалізації двох елементарних методів сортування.

Проведіть серію експериментів для оцінки продуктивності реалізацій для різних типів вхідних даних (малий/великий розмір; випадкові/частково відсортовані; без однакових значень/з малою кількістю однакових значень/з великою кількістю однакових значень).

За результатами експериментів зробіть висновки, які з факторів найсуттєвіше впливають на продуктивність досліджуваних алгоритмів сортування.

Порівняйте отримані результати з таблицями на с.268, 320, 347, 379, 436 [3].

Зробіть висновки, чи отримано практичне підтвердження лем про продуктивність сортування; для яких даних досліджувани алгоритми підходять якнайкраще, а у яких випадках використовувати їх недоцільно.

Варіанти завдань

- 1) сортування вибором, стандартний та покращений алгоритм;

- 2) сортування вставками, стандартний та покращений алгоритм;
- 3) бульбашкове сортування;
- 4) сортування методом Шелла для різних послідовностей h ;
- 5) сортування методом розподіляючого підрахунку.

Склад звіту практичної роботи

- постановка задачі (вказати, які алгоритми досліджуються);
- блок-схеми реалізацій, на яких виконано аналіз складності алгоритмів;
- результати дослідження у вигляді графіків в координатах: кількість елементів – час (елементи без повторень, без попереднього сортування); степінь відсортованості – час (при фіксованій загальній кількості елементів, без повторень); кількість однакових елементів – час (при фіксованій загальній кількості елементів та низькій попередній відсортованості).
- висновки про доцільність використання кожного з алгоритмів для типових вхідних даних та про відповідність результатів експериментального дослідження аналітичним оцінкам складності.

ПРАКТИЧНА РОБОТА №10

Тема: Алгоритми сортування. Методи сортування малих обсягів даних

Мета роботи: дослідження та порівняння алгоритмів сортування.

Запитання на допуск до роботи

1. Порівняйте елементарні методи сортування (сортування вибором, сортування вставками, бульбашкове сортування). Вкажіть, на яких вхідних даних ці методи працюють найкраще.
2. Порівняйте елементарні методи сортування (сортування вибором та сортування методом розподіляючого підрахунку). Вкажіть, на яких вхідних даних ці методи працюють найкраще.

3. Порівняйте елементарні методи сортування (непряме (по індексам або показникам) та пряме сортування вибором). Вкажіть, на яких вхідних даних ці методи працюють найкраще.

4. Перерахуйте та прокоментуйте переваги та недоліки сортування методом Шелла для різних послідовностей h . Вкажіть, на яких вхідних даних цей метод працює найкраще.

Завдання

Ознайомтесь з різновидами методів сортування (елементарні методи сортування с.249-298 [3]; швидке сортування с.299-329 [3], с. 235-243 [1], с.223-229[2]).

Напишіть програмні реалізації кількох елементарних методів сортування (можна взяти з попередньої роботи) та швидкого сортування. Проведіть серію експериментів для оцінки продуктивності реалізацій для різних типів вхідних даних (малий/великий розмір; випадкові/частково відсортовані; без однакових значень/з малою кількістю однакових значень/з великою кількістю однакових значень).

За результатами експериментів зробіть висновки, для яких даних досліджувані алгоритми підходять якнайкраще, а у яких випадках використовувати їх недоцільно.

Склад звіту практичної роботи

- постановка задачі (вказати, які алгоритми досліджуються);
- блок-схеми реалізацій, на яких виконано аналіз складності алгоритмів;
- результати дослідження у вигляді графіків в координатах: кількість елементів – час (елементи без повторень, без попереднього сортування); степінь відсортованості – час (при фіксованій загальній кількості елементів, без повторень); кількість однакових елементів – час (при фіксованій загальній кількості елементів та низькій попередній відсортованості).

- висновки про доцільність використання кожного з алгоритмів для типових вхідних даних та про відповідність результатів експериментального дослідження аналітичним оцінкам складності.

ПРАКТИЧНА РОБОТА №11

Тема: Алгоритми сортування. Методи сортування великих обсягів даних

Мета роботи: дослідження та порівняння алгоритмів сортування.

Запитання на допуск до роботи

1. Перерахуйте та прокоментуйте переваги та недоліки швидкого сортування. Які найпоширеніші модифікації цього методу сортування? Вкажіть, на яких вхідних даних цей метод працює найкраще.
2. Перерахуйте та прокоментуйте переваги та недоліки сортування злиттям. Які найпоширеніші модифікації цього методу сортування? Вкажіть, на яких вхідних даних цей метод працює найкраще.
3. Перерахуйте та прокоментуйте переваги та недоліки пірамідального сортування. Які найпоширеніші модифікації цього методу сортування? Вкажіть, на яких вхідних даних цей метод працює найкраще.

Завдання

Ознайомтесь з різновидами методів сортування (швидке сортування с.299-329 [3], с. 235-243 [1], с.223-229[2]; сортування злиттям с.330-354 [3], с. 116-128 [1]; пірамідальне сортування с.355-400 [3], с. 244-247 [1], с.229-240[2]).

Напишіть програмні реалізації цих методів сортування. Проведіть серію експериментів для оцінки продуктивності реалізацій для різних типів вхідних даних (малий/великий розмір; випадкові/частково відсортовані; без однакових значень/з малою кількістю однакових значень/з великою кількістю однакових значень).

За результатами експериментів зробіть висновки, які з факторів найсуттєвіше впливають на продуктивність досліджуваних алгоритмів сортування.

Порівняйте отримані результати з таблицями на с.268, 320, 347, 379, 436 [3].

Зробіть висновки, чи отримано практичне підтвердження лем про продуктивність сортування; для яких даних досліджувані алгоритми підходять якнайкраще, а у яких випадках використовувати їх недоцільно.

Склад звіту практичної роботи

- постановка задачі (вказати, які алгоритми досліджуються);
- блок-схеми реалізацій, на яких виконано аналіз складності алгоритмів;
- результати дослідження у вигляді графіків в координатах: кількість елементів – час (елементи без повторень, без попереднього сортування); степінь відсортованості – час (при фіксованій загальній кількості елементів, без повторень); кількість однакових елементів – час (при фіксованій загальній кількості елементів та низькій попередній відсортованості).
- висновки про доцільність використання кожного з алгоритмів для типових вхідних даних та про відповідність результатів експериментального дослідження аналітичним оцінкам складності.

ПРАКТИЧНА РОБОТА №12

Тема: Алгоритми сортування. Методи сортування даних з довгими ключами

Мета роботи: дослідження та порівняння алгоритмів сортування.

Запитання на допуск до роботи

1. Перерахуйте та прокоментуйте переваги та недоліки порозрядного сортування. Які найпоширеніші модифікації цього методу сортування? Вкажіть, на яких вхідних даних цей метод працює найкраще.
2. Перерахуйте відомі вам методи сортування спеціального призначення та вкажіть їх область використання.

Завдання

Ознайомтесь з різновидами методів порозрядного сортування (с.401-437 [3]). Напишіть програмні реалізації стандартного алгоритму порозрядного сортування, алгоритму LSD та алгоритму MSD; з попередньої роботи

візьміть також швидке сортування. Проведіть серію експериментів для оцінки продуктивності реалізацій для різних типів вхідних даних (малий/великий розмір; випадкові/частково відсортовані; без однакових значень/з малою кількістю однакових значень/з великою кількістю однакових значень).

За результатами експериментів зробіть висновки, які з факторів найсуттєвіше впливають на продуктивність досліджуваних алгоритмів сортування.

Порівняйте отримані результати з таблицями на с.268, 320, 347, 379, 436 [3].

Зробіть висновки, чи отримано практичне підтвердження лем про продуктивність сортування; для яких даних досліджувані алгоритми підходять якнайкраще, а у яких випадках використовувати їх недоцільно.

Склад звіту практичної роботи

- постановка задачі (вказати, які алгоритми досліджуються);
- блок-схеми реалізацій, на яких виконано аналіз складності алгоритмів;
- результати дослідження у вигляді графіків в координатах: кількість елементів – час (елементи без повторень, без попереднього сортування); степінь відсортованості – час (при фіксованій загальній кількості елементів, без повторень); кількість однакових елементів – час (при фіксованій загальній кількості елементів та низькій попередній відсортованості).
- висновки про доцільність використання кожного з алгоритмів для типових вхідних даних та про відповідність результатів експериментального дослідження аналітичним оцінкам складності.

ПРАКТИЧНА РОБОТА №13

Тема: Алгоритми пошуку в графі. Пошук в ширину та пошук в глибину

Мета роботи: дослідження алгоритмів та окреслення області їх застосування.

Запитання на допуск до роботи

1. Перерахуйте відомі вам алгоритми обходу вершин в графі та вкажіть їх область використання.
2. Прокоментуйте переваги та недоліки цих алгоритмів.

Завдання

Напишіть програмні реалізації двох алгоритмів, що розв'язують поставлену задачу. Проведіть серію експериментів для оцінки продуктивності реалізацій для різних обсягів вхідних даних.

За результатами експериментів зробіть висновки, для яких даних досліджувані алгоритми підходять якнайкраще, а у яких випадках використовувати їх недоцільно.

Склад звіту практичної роботи

- постановка задачі (вказати, які алгоритми досліджуються);
- результати експериментів у вигляді графіків;
- висновки та рекомендації щодо застосування досліджених алгоритмів.

ПРАКТИЧНА РОБОТА №14

Тема: Алгоритми пошуку в графі. Пошук найкоротших шляхів у зваженому графі

Мета роботи: дослідження алгоритмів та окреслення області їх застосування.

Запитання на допуск до роботи

1. Перерахуйте відомі вам алгоритми пошуку найкоротших шляхів у зваженому графі та вкажіть їх область використання.
2. Прокоментуйте переваги та недоліки цих алгоритмів.

Завдання

Напишіть програмні реалізації двох алгоритмів, що розв'язують поставлену задачу. Проведіть серію експериментів для оцінки продуктивності реалізацій для різних обсягів вхідних даних.

За результатами експериментів зробіть висновки, для яких даних

досліджувані алгоритми підходять якнайкраще, а у яких випадках використовувати їх недоцільно.

Склад звіту практичної роботи

- постановка задачі (вказати, які алгоритми досліджуються);
- результати експериментів у вигляді графіків;
- висновки та рекомендації щодо застосування досліджених алгоритмів.

ПРАКТИЧНА РОБОТА №15

Тема: **Алгоритми пошуку в графі. Пошук найкоротших шляхів у графі, заданому в координатній сітці**

Мета роботи: дослідження алгоритмів та окреслення області їх застосування.

Запитання на допуск до роботи

1. Перерахуйте відомі вам алгоритми пошуку найкоротших шляхів у графі, заданому в координатній сітці, та вкажіть їх область використання.
2. Прокоментуйте переваги та недоліки цих алгоритмів.

Завдання

Напишіть програмні реалізації двох алгоритмів, що розв’язують поставлену задачу. Проведіть серію експериментів для оцінки продуктивності реалізацій для різних обсягів вхідних даних.

За результатами експериментів зробіть висновки, для яких даних досліджувані алгоритми підходять якнайкраще, а у яких випадках використовувати їх недоцільно.

Склад звіту практичної роботи

- постановка задачі (вказати, які алгоритми досліджуються);
- результати експериментів у вигляді графіків;
- висновки та рекомендації щодо застосування досліджених алгоритмів.

ПРАКТИЧНА РОБОТА №16

Тема: Алгоритми пошуку в графі. Топологічне сортування

Мета роботи: дослідження алгоритмів та окреслення області їх застосування.

Запитання на допуск до роботи

1. Перерахуйте відомі вам алгоритми пошуку топологічного сортування вершин орієнтованого графа та вкажіть їх область використання.
2. Прокоментуйте переваги та недоліки цих алгоритмів.

Завдання

Напишіть програмні реалізації двох алгоритмів, що розв'язують поставлену задачу. Проведіть серію експериментів для оцінки продуктивності реалізацій для різних обсягів вхідних даних.

За результатами експериментів зробіть висновки, для яких даних досліджувані алгоритми підходять якнайкраще, а у яких випадках використовувати їх недоцільно.

Склад звіту практичної роботи

- постановка задачі (вказати, які алгоритми досліджуються);
- результати експериментів у вигляді графіків;
- висновки та рекомендації щодо застосування досліджених алгоритмів.

Література

1. Ахо А., Хопкрофт Д., Ульман Д. Структуры данных и алгоритмы / М.: Вильямс, 2000 – 384с.
2. Гудман С., Хидетниemi С. Введение в разработку и анализ алгоритмов/М.: Мир, 1981 – 368 с.
3. Седжвик Р. Фундаментальные алгоритмы на C++ / К.: ДияСофт, 2001 – 688с.
4. Ковалюк Т.В. Основы програмування / К.: BHV, 2005 – 384 с